REMARKS

This amendment is filed in response to the office action mailed 08 August 2008 and is accompanied by a Petition for Three-Month extension of time and the requisite fee (and/or authorization to charge the requisite fee) to extend the due date to 08 February 2009. Entry of the proposed amendment and reconsideration with an eye toward allowance is respectfully requested.

Applicant acknowledges that the prior amendments to claims 14 and 28 have overcome the previous objections to those claims and that those objections have been withdrawn.

Claims 7-9, 14-18, and 24-30 were pending in the application prior to entry of this amendment. Claims 14 and 27-28 have been cancelled without prejudice and claims 31-47 have been added.

Remarks Concerning 35 USC §112 and 35 USC §102/103 Prior Art Rejections

Applicant wishes to thank the examiner for spending the time reading the specification as filed including reviewing the various drawings and communicating his and the office's thoughts relative to the claimed invention and the cited prior art.

Applicant herewith addresses some of the office's concerns regarding such elements as "shared memory" versus "shared data", "distributed computing" versus "distributed shared memory" or "distributed shared memory arrangement", "global or shared memory" versus "local independent memory", as well as characteristics of an application program written to operate on a single computer versus a computer program written to operate on multiple computers.

Firstly, Applicant respectfully disagrees with the examiner's suggestion that the limitation "without forming a distributed shared memory arrangement" added in the last

amendment is not being supported by the disclosure as filed. Applicant submits that this limitation is inherent and implicit in the disclosure as filed.

Applicant, for example, notes that the portion of the specification which discusses FIG. 5 and refers to each as machine having a memory capability of 10MB, refers to "a shared memory capability" and this may initially cause some confusion until the full description of the different examples of the invention are considered. The application also refers to an "unshared" memory capability of the or each machine in the second paragraph of the Detailed Description at page 6. This possible initial confusion, if any, may result from an evolving use of the term "shared memory" and its changed meaning in the art over the years. The phrase shared memory in the application as filed refers to portions of the local memory in a single machine or computer that store commonly referenced memory fields or variables in or from the application program code. These "shared memory" locations are not accessible by the processors of different ones of the computers, but instead are replicated memory locations or fields. The updating of these commonly referenced memory locations or fields is accomplished over a communications link that in at least one claimed example is an intranet or the Internet, and therefore an entirely different type of communications link than for example a high-speed internal or external bus.

In this regard, applicant further submits that the office may have a mis-understanding of the differences between the two different concepts of "distributed computing" and "distributed shared memory".

"Distributed computing" takes place where there is a plural number of computers that are interconnected in some way (e.g. "interconnected computers") and the processor of each separate computer carries out some computation. Thus the computation takes place at different locations or at least in different separate computers and is said to be "distributed".

With such "interconnected computers" there is also memory available at each location. Each computer memory conceptually has two parts. A first part of each computer memory is for the local operating system. A second part of each computer memory is available for the application program.

Applicant further submits, that in the prior art "distributed shared memory system (DSM)" at least the second part of each computer's memory is able to be directly accessed (by reading and writing to the memory) by all other computers. So if there are "n" identical computers and the size of the second part (e.g. the application program part) of each computer's memory is 10MB, then the total available memory able to be accessed by each, any, and every machine is essentially n times the 10MB of the each computer or 10nMB. This means that the second part of each computer's memory is not independent because the other computers can read from, and write to, this memory.

This gives in the prior art the advantage of a large volume of memory storage but the disadvantage is that it takes a long time for a first computer to read and receive data from a memory location (called a field in the JAVA language) physically located in a second computer. In particular, the processor of the first machine or computer must wait (and therefore delay ongoing execution), whilst this memory read process takes place. It also follows that in a prior art "distributed shared memory system (DSM)" arrangement there is no duplication or replication of memory and thus no requirement for memory updating. Any memory location (or JAVA field) is stored only in a single location and that location may be on any of the plurality of separate computers.

By comparison, the claimed embodiments of the present invention are based upon an entirely different concept. For the "n" interconnected computers each having a second memory portion of 10MB, no computer can directly access or read the 10MB of any other computer. Therefore in at least some embodiments, the memories of the different

computers are independent memories. Other differences from the cited prior are also recited.

For this reason the memory in at least some of the amended and new independent claims (e.g., Claim 31) is specified as being independent. There is thus no sharing of memory locations between the multiple computers, but there is a sharing of data. Instead of storing a particular item of data only once and in only one of the 10nMB shared memory locations, each memory location (or field) created in the independent memory of one particular computer during that particular computer's processing, is replicated in the independent memory of each and every other computer, and the particular data item is replicated in that replicated independent memory as well (at least for memory locations or fields that are commonly referenced in the application programs executing on the different computers).

It is therefore necessary to update every other different computer with any change made to a local memory location by any one computer. This updating represents an overhead operation that might suggest disadvantage or inefficiency associated with the updating approach. But the inventive arrangement means that each and every computer can satisfy its memory reading requirement from its own local independent memory (which is very quick). Since application programs in general read data very much more frequently then they write new data, the stratagem results in a considerable increase in overall operating speed. In at least some claimed embodiments that describe details of the program modification, the element of "detecting those instructions which write to, or manipulate the contents of, any of said listed memory records" is recited as these are the instructions that might result in a change of the value or contents of that memory location or field, and that therefore may require updating. On the other hand, the modification of the program is not required for instructions that merely read the contents of, any of said listed memory records. These aspects are described in the application as filed in the second and third paragraphs of page 8 and in the claims as originally filed.

In the present invention, the data (but not the memory location) may be regarded as being "shared" in the sense that a computational result created by, say computer M3 as described, is written into a memory location of the 10MB memory of computer M3. This computational result (e.g., rewriting or manipulation) is then propagated to all other computers, and stored in their own corresponding replicated local memory locations. Therefore, as a result of the update or propagation of the computational result from M3, computer M4 is now able to locally read from its own local independent memory the result of the computation carried out by computer M3. Thus the data created by computer M3 is shared with computers M1, M2, M4, ..., Mn even though the memory locations are **not** shared.

In the early days of computing the term "shared memory" referred to memory to which more than one processor had access (to read and/or to write). However, in recent times this term has become blurred to refer to data in memory which is available to the application program running on the computer system. Thus the term "shared memory" has come to acquire a meaning that the data is accessible to all computers (whether this is achieved either by making various memories accessible by all computers, or by replicating or duplicating data in memories which are not accessible by all computers).

Applicant submits that it is clear from the description of the invention in the disclosure as filed, the phrase "shared memory" refers to two or more computers which cannot read from, or write to, each other's memory, and in which memory locations are both replicated and updated with the shared data, values, or other content as described. Because of the duplication or replication of the data for commonly referenced memory fields in multiple locations this may also be referred to as "replicated shared memory" or "replicated shared data" and such duplication or replication of the data in the local memories of the different machines or computers is clearly described in the application as filed.

Applicant submits that when the application as filed is considered as a whole and when the different descriptive portions are read by a person having ordinary skill in the art, the first paragraph on page 6 of the application as filed should be interpreted in the following way., where the underlined portions indicate additions to the description that are taken from other portions of the application as filed and/or are implied from or inherent in the description.

> As a consequence of the above described arrangement, if each of the machines M1, M2 ... Mn has, say, a replicated shared memory capability of 10MB, then the total replicated shared memory available to each application 50 is not, as one might expect, 10nMB but rather only 10MB. Changes made to a replicated memory location in one machine are propagated to the corresponding replicated memory locations in all other machines. Thus the other machines are able to share the result of the calculation of the one machine. In this sense the contents of the replicated memory locations are shared. However, how this results in improved operation will become apparent hereafter. Naturally, each machine M1, M2 ... Mn has an unshared and unreplicated memory capability. The unshared and unreplicated memory capability of the machines M1, M2 ... Mn are normally approximately equal but need not be. Typically the unshared and unreplicated memory of machine Mn holds the operating system of machine Mn.

Although applicant submits that the application fully describes the invention as required by 35 USC §112, applicant might propose to amend the specification to consolidate the description already contained in the application as filed with this language so that in reading this initial and succinct description, the invention is more readily appreciated.

Applicant submits, that among the other features and benefits, this structure and method enables an application program written to operate on a single computer with the code thread structure and modification described to execute simultaneously on the multiple computers of the multiple computers system, with each computer able to locally access the replicated data.

Although a high-speed and high-bandwidth communications link is not precluded from use, the generally lower frequency or occurrence of writing, re-writing, or manipulating data in the memory locations relative to the frequency of reading such data means that a relatively low-bandwidth or low-speed communications link may be used for the updating without significantly slowing overall performance. Therefore certain non-limiting embodiments may use an intranet, the Internet, a local area network, or other network for the propagation and updating, without imposing a significant performance penalty.

Applicant also wishes to point out that the Disclosure in the Annexure of the application as filed provides example program listings for a JAVA implementation that provides further description of the code modification, the involvement of the alert and the DRT in the communication, the sending of the name tag and value of the field over the network, and the receiving by a different DRT of the name tag and value, as well as the writing of the value from the network into the field location in memory of the receiving DRT of the receiving computer. Applicant does not believe this additional disclosure from the Annexure is needed to enable the invention as claimed but has amended the Detailed Description to include this JAVA code example description in the Detailed Description.

35 USC 102 and 35 USC 103 rejections

In the present office action, the examiner has reviewed the prior rejections as well as applicant's response to it. For example, the examiner has reviewed the rejection of the prior pending independent claims 7 and 14 on Chen, suggesting that Chen teaches an

application written to operate on only a single computer. The examiner also reviewed the prior rejection suggesting that Morshed teaches distributed computation. The examiner further suggests in light of applicant's earlier arguments for the prior pending claims, that with respect to Chen, "the fact that Chen discloses a distributed memory system does not preclude an application which runs on it from being written to operate only on a single computer".

In the present office action, Claims 7, 8, 24 and 27 are under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent 6,760,903 to Morshed et al. (hereinafter "Morshed") in view of "MultiJav: A Distributed Shared Memory System Based on Multiple Java Virtual Machines" by Chen et at. (hereinafter "Chen").

Claim 9 is rejected under 35 U.S.C. 103(a) as being unpatentable over Morshed and Chen in view of U.S. Patent No. 5,802,585 to Scales et al. (hereinafter "Scales").

Claims 14, 15, 26, and 28 are rejected under 35 U.S.C. 103(a) as being unpatentable over Scales in view of Chen.

Claim 16 is rejected under 35 U.S.C. 103(a) as being unpatentable over Scales and Chen as applied to claim 14 above, and further in view of Morshed.

Claims 17 and 18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Scales and Chen as applied to claim 14 above, and further in view of U.S. Patent Application Publication 2004/0163077 by Dimpsey et al. (hereinafter "Dimpsey.")

Claim 25 is rejected under 35 U.S.C. 103(a) as being unpatentable over Morshed and Chen as applied to claim 7 above, and further in view of U.S. Patent No. 6,862,608 to Buhlman et al. (hereinafter "Buhlman").

Without admitting the propriety of the rejection, applicant has significantly amended each of the pending independent claims or replaced the independent claims with newly written independent claims. These amended and/or rewritten independent claims include changes that address the differences between the cited prior art and also address the examiner's 35 USC §112 issues.

For example, dependent claim 7 (Claim 8, 24-25, and 29-30 are dependent from claim 7), and has cancelled without prejudice independent claims 14 and 27, thereby mooting the present 35 U.S.C. 103(a) rejection of these claims. Claim 28 also dependent from claim 7 has also been cancelled without prejudice. Claim 9 is dependent from independent claim 7 and in the alternative from dependent claim 8. Claim 14 has been canceled, and the claims formerly dependent therefrom, including claims 15-18 and 26 are amended to depend from new independent claim 31.

Applicant respectfully submits that each of independent claims 7, 31, and 45 are now patentably distinguished over the cited prior art and that claims dependent from these independent claims are also patentable over the cited art for at least the same reason as the underlying independent claim and further because each adds additional distinguishing limitations.

Claim 7 as amended provides further description of the structure in which the claimed embodiment of the method operates. Applicant's remarks from page 15-19 are also incorporated here (without repeating them) to distinguish the nature of the program written to operate on a single computer and the nature of the local memory that is not shared even though the data is shared.

For example, claim 7 now requires "the application program having application program code including a plurality of code threads all intended to execute on and reference a single computer having a single processing unit or symmetric multiple processing units and a single independent local memory with a local memory capacity that is not shared with any other single computer of said plurality of single computers".

Claim 7 now also requires, that "different portions of said application program being simultaneously executable on each different one of the plurality of single computers

with each different one of the plurality of single computers having a different independent local memory accessible only by a corresponding portion of the application program" .

Applicant submits that none of the cited references either alone or in any combination disclose, suggest, or motivate any need for these feature either alone or in combination with the other requirements of the claim.

Claim 9, dependent from claim 7 further requires particular additional steps for the modifying or the application program, including:

> (i) detecting instructions in the unmodified application program which reference the same common memory records,
>
> (ii) listing all such commonly referenced memory records and providing a naming tag for each said listed commonly referenced memory record,
>
> (iii) detecting those instructions which write to, or manipulate the contents of, any of said listed commonly referenced memory records, and
>
> (iv) generating an alert instruction following each said detected commonly referenced memory record write or manipulate instruction, said alert instruction forwarding the re-written or manipulated contents and name tag of each said re-written or manipulated listed memory record.

Applicant again submits that none of the cited references either alone or in any combination disclose, suggest, or motivate any need for these steps associated with modifying the application program either alone or in combination with the other requirements of the claim, particularly in view of the further requirements of underlying method claim 7.

Independent claim 31, is newly submitted and is directed to a method of compiling or modifying an application program. It further defined the structure of the application program as a program:

> "written to include a plurality of instruction code threads intended to execute on and reference only a single computer having a single central processing unit (CPU) or symmetric multiple processing units and a single independent local memory that is not shared with any other computer of said plurality of single computers but to run simultaneously on each one of a plurality of computers interconnectable via a communications link, with different portions of said application program being simultaneously executable on different ones of said plurality of single computers with each one of the plurality of single computers having the independent local memory accessible only by the corresponding portion of the application program"

Claim 31 further requires particular steps associated with the operation of the program, including:

> (i) detecting instructions in the unmodified application program which reference the same common memory records;
>
> (ii) listing all such commonly referenced memory records and providing a naming tag for each said listed commonly referenced memory record;
>
> (iii) detecting those instructions which write to, or manipulate the contents of, any of said listed commonly referenced memory records; and
>
> (iv) generating and inserting an alert instruction into the unmodified application program to create the modified application program for handling by a distributed run time (DRT) following each said detected commonly referenced memory record write or manipulate instruction indicating that the contents or value of the commonly referenced memory record were re-

written or manipulated and may have changed during execution of a code thread, said alert instruction being operative for initiating propagation of the re-written or manipulated contents and name tag of each said re-written or manipulated listed commonly referenced memory record via the communications link to the distributed run times (DRTs) of each other of the single computers.

Applicant again submits that none of the cited references either alone or in any combination disclose, suggest, or motivate any need for these steps associated with a method of compiling or modifying an application program either alone or in combination with the other requirements of the claim.

Finally, new independent claim 45 is directed to a method of loading an original application program onto each of said plurality of single computers in a multiple computer system. Applicant submits that claim 45 requires several features that are not disclosed, suggested, or motivated by the cited prior art. These include the following requirements:

Claim 45 requires that:

> ... the original application program having original application program code including a plurality of original code threads all written to execute on and reference a single computer having a single processing unit or symmetric multiple processing units and a single local memory with a local memory capacity that is not shared with any other single computer of said plurality of single computers, ...

Claim 45 also requires:

> ... the original application program being modified to form at least one modified application program with different portions of said modified application program being simultaneously executed within a

different independent local processor and a different independent local memory within each different one of the plurality of single computers, said different independent local memory within each said different single computer not forming a distributed shared memory arrangement and being accessible during execution of said application program and said different portions of said application program only by the different portion of the application program actually executing within the different local processing unit or symmetric multiple processing units of the different computer, ...

Claim 45 further requires particular loading, and modifying procedures as follows:

... loading the application program onto each different computer of said plurality of single computers, said application program including a reference to a program memory field that may be references by one or more of said plurality of computers during execution of their respective different portion of the application program; and

modifying the application program on each said different single computer before execution of said different portion of the application program on each said different single computer; and

said modification of the application program includes an insertion of at least one code thread prior to execution that upon execution by one of said single computers initiates a sequence of events that result in a network packet communication over said Internet or intranet network communications link that contains an identifier of the referenced memory field and the contents or value of that memory field.

In light of these differences, Applicant again submits that none of the cited references either alone or in any combination disclose, suggest, or motivate any need for these steps associated with modifying the application program either alone or in

combination with the other requirements of the claim, particularly in view of the further requirements of underlying method claim 45.

In view of these amended and new independent claims, and the general remarks distinguishing applicant's architecture, methods, operations, and procedures from the cited art as well as the particular claim elements identified above, Applicant submits that each of the now pending independent claims is patentable over the cited art and that the rejections applied cannot stand and should be withdrawn. Applicant further submits that the 35 USC 112 based rejection should be withdrawn in light of the amended and new claims and further in view of the remarks contained herein.

Finally, applicant submits that each of the dependent claims is patentable over the cited art for at least the same reasons as the underlying base claim and further because each adds additional distinguishing limitations. Applicant has hesitated in reciting each different and additional feature into these remarks in the interest of limiting the length of an already long response to office action. However, applicant does wish to direct the examiner's attention to several new features that are recited in newly added dependent claims 32-44 and 46 that were not examined in the previous office actions.

CONCLUSION

In view of the above amendment, applicant believes the pending application is in condition for allowance.

Applicant believes no fee is due with this response other than those which have already been paid. However, if a fee is due, including any fees due as a result of the Petition for Extension of Time (3 Months) or for added claims, or otherwise, please charge our Deposit Account No. 50-2207, under Order No. 61130-8110.US01 from which the undersigned is authorized to draw.

Dated: 14 May 2009

Respectfully submitted,

By R. Michael Ananian

R. Michael Ananian
Registration No.: 35,050

PERKINS COIE LLP
P.O. Box 1208
Seattle, Washington 98111-1208
(650) 838-4300
(650) 838-4350 (Fax)
Attorney for Applicant

Customer No.: 22918

Certificate of Transmission

I hereby certify that this paper is being deposited via the United States Patent and Trademark Office, Electronic Filing Systems (EFS), addressed to the attention of: MS Amendment, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

Dated: May 15, 2009 Signature: Cathi L.G. Thoorsell
(Cathi L.G. Thoorsell)